REMARKS

Reconsideration of this application, as amended, is respectfully requested. The following remarks are responsive to the Office Action mailed September 23, 2001.

Amendment of Claims

Claims 1 and 9, have been amended to include subject matter corresponding substantially to that of cancelled claims 6 and 18 respectively.

Claim 21, has been amended to replace the word "systematic" with the word "symmetric." Support for this amendment may be found in the title, abstract, and on pages 6, 10, 22 and Figure 9 of the specification.

The above amendments are responsive to the Office Action of September 23, 2002, and place the respective claims in condition for allowance.

Response to Claim Rejections – 35 USC § 112

Claim 21 stands rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicants regard as the invention. Specifically, the Office Action indicates that the phrase "systematic multiprocessor environment", as it appears in claim 21, is not included in the specification.

As indicated above, Claim 21 has been amended to replace the word "systematic" with the word "symmetric".

Response to Claim Rejections – 35 USC § 102

Claims 1, 2, and 9-11 stand rejected under 35 U.S.C. 102(e) as being allegedly anticipated by U.S. Patent No. 6,167,423 (hereinafter Chopra).

To anticipate a claim, the reference must teach every element of the claim.

> "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference."

Applicants respectfully submit the rejection of claims 1 and 9 under 35 U.S.C. § 102(e) has been overcome and is addressed for the reason that Chopra does not disclose each and every limitation of claims 1 and 9, as amended, of the present application.

Claim 1, as amended, includes the following limitation:

identifying a processor affinity attributed to the task....

The Office Action in, rejecting claim 1, contends that Chopra does not teach the above limitations (Office Action, #24).

Therefore, Chopra does not disclose each and every limitation of amended claim 1, as required to support a rejection of this claim under 35 U.S.C. § 102(e).

Independent claim 9 includes a limitation corresponding substantially to the above-discussed limitation of claim 1. The above remarks are accordingly also applicable to a consideration of this independent claim. Accordingly, Applicants request that the above remarks and amendments contained herein also be considered when examining independent claim 9 for allowability.

As dependent claims are deemed to include all limitation of claims from which they depend, the rejection of claims 2, and 10-11 under 35 U.S.C. 102(e) is also addressed by the above remarks, and the amendments contained herein.

Response to Claim Rejections – 35 USC § 103

Claims 3 and 16 stand rejected under 35 U.S.C. 103(a) as being unpatenable over Chopra in view of U.S. Patent No. 6,314,089 (hereinafter Szlam).

Claims 4, 12, 13, 17, and 21-23 stand rejected under 35 U.S.C. 103(a) as being unpatenable over Chopra in view of U.S. Patent No. 5,327,557 (hereinafter Emmond).

Claim 5 stands rejected under 35 U.S.C. 103(a) as being unpatenable over Chopra in view of U.S. Patent No. 6,222,530 (hereinafter Sequeira).

Claims 6 and 18 stand rejected under 35 U.S.C. 103(a) as being

unpatenable over Chopra in view of U.S. Patent No. 6,161,688 (hereinafter Wipfel).

Claims 7, 8, 19, and 20 stand rejected under 35 U.S.C. 103(a) as being unpatenable over Chopra in view of U.S. Patent No. 6,223,207 (hereinafter Lucovsky).

Claims 14 and 15 stand rejected under 35 U.S.C. 103(a) as being unpatenable over Chopra in view of Emmond in further view of Sequeira.

Applicants submit that the rejections of claims 3, 16, 4, 12, 13, 17, 21-23, 5, 6, 18, 7, 8, 19, 20, 14 and 15 under 35 U.S.C. § 103 are groundless for the reason that the prior art references when combined do not teach or suggest all of the claim limitations of the independent claims of the present application. Turning to claim 6.

> To establish a **prima facie** case of **obviousness**, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on applicant's disclosure.

Claim 6 includes the following limitation:

a processor affinity attributed to the task....

The Office Action, in rejecting claim 6, contends that the above limitation is anticipated by the following disclosure in Wipfel:

> Although one embodiment of the invention provides each processor 216 with its own local resource queue 702, in other embodiments some processors 216 have no resource queue 702. In some embodiments, a local queue 702 is associated with a set of processors 216 rather than a single processor 216, with a set of processes or tasks or threads, and/or with a set of one or more cluster nodes 106.

Wipfel, Col. 16, lines 40-47.

The above quote from Wipfel describes three embodiments with respect to processors and local resource queues. The first embodiment describes each processor with its own local resource queue. The second embodiment describes some processors without local resource queues. The third embodiment describes multiple processors associated with a single local resource queue. Further, the above quote from Wipfel describes that a local resource queue is associated with a set of processes or tasks or threads, and/or with a set of one or more cluster nodes.

Claim 6 requires a method whereby a processor affinity is attributed to a task. For example, an affinity of a first task for a first processor may be attributed to the first task and an affinity of a second task for a second processor may be attributed to the second task. In contrast, Wipfel does not describe a processor affinity that is attributed to a task; rather Wipfel

describes alternate embodiments of processors and local resource queues. Indeed, the above quote from Wipfel describes a resource queue associated with a task; however, the above quote from Wipfel does not describe a task as having an affinity for a processor much less an affinity that may be attributed to the task.

Independent claims 1 and 9, as amended, each include a limitation corresponding substantially to the above-discussed limitation of cancelled claim 6. The above remarks are accordingly also applicable to a consideration of these independent claims.

In summary, Chopra in combination with Wipfel does not teach or suggest each and every limitation of amended claims 1 and 9 as required to support rejection of theses independent claims of the present application under 35 U.S.C.§ 103.

In addition, if an independent claim is nonobvious under 35 U.S.C. § 103(a) then, any claim depending therefrom is nonobvious and the rejections of claims 3, 16, 4, 12, 13, 17, 5, 7, 8, 19, 20, 14 and 15 claims under 35 U.S.C. § 103(a) are also addressed by the above remarks.

Claim 21, as amended, requires:

> tasks utilizing a pool of threads executable within a symmetric multiprocessor environment.

The Office Action, in rejecting claim 21, does not point out any specific disclosures in Emmond that are anticipatory. Applicants do however note that Emmond discloses the following:

> The terminal and menu processes 17 and the application data base and computation processes 18 could be performed by separate data processors, or they could be performed by a single data processor having an operating system that supports a multiprocessing environment.

Col. 4, lines 55-60.

The above quote from Emmond describes that a process (i.e., a terminal, menu, application database or computation) may be performed by separate data processors or by a single data processor. Further, performance by a single data processor requires an operating system that supports a multiprocessing environment (i.e., supporting concurrent execution of multiple processes).

Claim 21 requires tasks utilizing a pool of threads that execute within a symmetric multiprocessor environment (i.e., all processors controlled by a single operating system[1]). In contrast, Emmond does not disclose tasks utilizing a pool of threads executable within a symmetric multiprocessor environment because the above quote from Emmond describes processes that are performed by separate data processors without regard to the operating system. Indeed, the above quote from Emmond describes a

multiprocessing environment; however, a multiprocessing environment is not a multiprocessor environment. To be sure, Emmond describes "a single processor having an operating system that supports a multiprocessing environment." Emmond therefore cannot be said to anticipate the above quoted limitation because Emmond describes processes that are performed by separate data processors without regard to the operating system and claim 21 requires tasks utilizing a pool of threads that execute within a symmetric multiprocessor environment.

If an independent claim is nonobvious under 35 U.S.C. § 103 then, any claim depending therefrom is nonobvious and rejection of claims 22-23 under 35 U.S.C. § 103 is also addressed by the above remarks.

In summary, Chopra in combination with Emmond does not teach or suggest each and every limitation of claims 21-23 as required to support rejections of the independent claims of the present application under 35 U.S.C.§ 103.

In summary, Applicants believe that all rejections presented in the Office Action have been fully addressed and withdrawal of these rejections is respectfully requested. Applicants furthermore believe that all claims are now in a condition for allowance, which is earnestly solicited.

---

[1] Application page 10, line 10-12.

If the Examiner believes a telephone interview would expedite the prosecution of this application, the Examiner is invited to contact André L. Marais at (408) 947-8200 x204.

If there are any additional charges, please charge them to Deposit Account No. 02-2666.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN

Dated: 01/23/2003

André L. Marais
Registration No. 48,095

12400 Wilshire Blvd.
Seventh Floor
Los Angeles, CA 90025-1026
(408) 947-8200

VERSION WITH MARKINGS TO SHOW CHANGES MADE

A marked up version of the claims is provided below. Additions are indicated with "___" and deletions are indicated within "[ ]."

1. (Amended) A method of executing a transaction task within a transaction processing system, the method including:

responsive to an event, identifying a workflow associated with the event; [and]

distributing a task, that at least partially executes the workflow, to an available thread within a pool of threads operating within a multiprocessor system;

identifying a processor affinity attributed to the task; and

assigning the available thread to a processor within the multiprocessor system according to the processor affinity attributed to the task.

2. (Unamended) The method of claim 1 wherein the event comprises a transaction event and the task comprises a transaction task responsive to a transaction request associated with the transaction event.

3. (Unamended) The method of claim 2 wherein the transaction task comprises a transaction routing task that routes the transaction request

associated with the transaction event to an agent of the transaction processing system.

4. (Unamended) The method of claim 2 within the transaction task comprises a transaction information task to either store or retrieve information pertinent to a transaction.

5. (Unamended) The method of claim 1 wherein the task has a real-time priority and is distributed in accordance with the real-time priority to the available thread within the pool of threads.

[6. (Cancelled) The method of claim 1 including identifying a processor affinity attributed to the task, and assigning the available thread to a processor within the multiprocessor system according to the processor affinity attributed to the task.]

7. (Unamended) The method of claim 1 including assigning the available thread to a processor within the multiprocessor system according to a thread priority.

8. (Unamended) The method of claim 7 including assigning the thread priority to the available thread based on a priority of the task distributed to the available thread.

9. (Amended) Apparatus for executing a transaction task within a transaction processing system, the apparatus comprising:

a dispatcher to identify a workflow associated with an event; and

a thread within a pool of threads operating within a multiprocessor system to execute a task that at least partially executes the workflow associated with the event[;] ,

the dispatcher to identify a processor affinity attributed to the task, and to assign the thread to a processor within the multiprocessor system according to the processor affinity attributed to the task.

10. (Unamended) The apparatus of claim 9 wherein the dispatcher generates the task that at least partially executes the workflow.

11. (Unamended) The apparatus of claim 10 including a task queue to which the task is dispatch by the dispatcher, and from which the thread within the pool of threads receives the task.

12. (Unamended) The apparatus of claim 11 including a scheduler that issues the task from the task queue to the thread within the pool of threads.

13. (Unamended) The apparatus of claim 12 wherein the scheduler issues the task from the task queue to the thread within the pool of threads based on a priority associated with the task.

14. (Unamended) The apparatus of claim 13 wherein the scheduler issues the task from the task queue according to a priority dynamically assigned to the task.

15. (Unamended) The apparatus of claim 13 wherein the scheduler issues the task from the task queue according to a real-time priority assigned to the task.

16. (Unamended) The apparatus of claim 9 wherein the task comprises a transaction routing task that routes a transaction request associated with the event to an agent of the transaction processing system.

17. (Unamended) The apparatus of claim 9 within the task comprises a transaction information task to either store or retrieve information pertinent to a transaction.

[18. (Cancelled) The apparatus of claim 9 including a dispatcher to identify a processor affinity attributed to the task, and to assign the thread to a processor within the multiprocessor system according to the processor affinity attributed to the task.]

19. (Unamended) The apparatus of claim 9 including to assign the thread to a processor within the multiprocessor system according to a thread priority.

20. (Unamended) The apparatus of claim 19 including assigning the thread priority to the thread based on a priority of the task distributed to the thread.

21. (Amended) A method of operating a transaction processing system employing a multiprocessor architecture, the method including:

establishing a queue of tasks, the queue of tasks including tasks for both system and transactional functions, and

servicing the queue of tasks utilizing a pool of threads executable within a [systematic] <u>symmetric</u> multiprocessor environment.

22. (Unamended) The method of claim 21 wherein the tasks for the system

functions include any one of reporting, administration or maintenance tasks performed within the transaction processing system.

23. (Unamended) The method of claim 21 wherein the tasks for the transactional functions include any one of routing, transaction data storage or transaction data retrieval tasks performed to facilitate a transaction within the transaction processing system.